GWD-C (NMWG Internal Draft)
Network Measurements Working Group
http://www-didc.lbl.gov/NMWG/
http://www-didc.lbl.gov/NMWG/metrics.pdf

Bruce Lowekamp
College of William and Mary
Brian Tierney
Lawrence Berkeley National Lab

January 29, 2002

Network Metrics for Grid Applications and Services
DRAFT

Status of this memo

This memo provides information to the Grid community regarded current practices of metrics and measurements used for network performance. It does not define any standards or make recommendations. Distribution is unlimited.

1 Introduction

This document catalogs and describes metrics of network performance that are useful to grid applications. The goals of this document are to:

- describe the metrics of interest to grid applications
- provide information about existing measurement techniques used for each metric
- define mappings between available tools and the measurements they implement
- define a dictionary of network performance terms for grid tools exchanging data, especially those using the GMA
- discuss issues in measuring each metric

This document focuses on existing and currently used tools and metrics. It does not attempt to define new standards or to define only the best metrics to use for grid applications. It does attempt to point out the advantages and disadvantages of each metric.

The NMWG is closely related to the IETF Internet Protocol Performance Metrics (IPPM) WG, with the exception that their focus is on defining best-practices metrics of use to network engineers, whereas the NMWG and this document focus on existing practices of grid application and grid-related tools. For consistency we will use much of the terminology defined in the IPPM Framework [6], although due to the different goals of NMWG and IPPM, certain sections of that Framework do not apply to this document.

The NMWG focuses on network metrics, but many metrics of network performance are necessarily limited by bottlenecks at the hosts making the measurements. We work to identify these limitations in this document, but a more thorough approach to such bottlenecks is the focus of the Internet2 End-to-end Performance WG.

2 Sample Grid use of network metrics

As an example of how networks metrics could be used in a Grid environment, we present an example of a Grid file transfer service. Assume that a Grid Scheduler [3] determines that a copy of a given file needs to be copied to site A before a job can be run. Several copies of this file are registered in a Data Grid Replica

Catalogue [1], so there is a choice of where to copy the file from. The Grid scheduler needs to determine the optimal method to create this new file copy, and to estimate how long this will take. To make this selection the scheduler must have the ability to answer these questions:

- what is the best source (or sources) to copy the data from?
- should parallel streams be used, and if so, how many?

Answering these questions accurately requires measurement and prediction of *available bandwidth* (both end-to-end and hop-by-hop), *latency*, *loss*, and others..

To determine which source to copy the data from requires bandwidth information on the end-to-end path between the destination and all possible sources.

Determining whether there would be an advantage in splitting up the copy, and, for example, copying the first half of the file from site B and in parallel the second half of the file from site C, requires hop-by-hop link availability information for each network path. If the bottleneck hop is a hop that is shared by multiple paths then there is no advantage to splitting up the file copy in this way.

Parallel data streams will usually increase the total throughput on uncongested paths, as shown by Hacker et al [4]. However on congested links, using parallel streams may just make the problem worse. Therefore metrics for loss are needed to determine how many parallel streams to use.

To compute a reliable estimate for the transfer time may require a combination of available bandwidth, loss, and delay information.

3 Metrics and Measurements

Before discussion and definition of network measurements can begin, we first define the relevant terms. The different research backgrounds of people participating in GGF necessarily bring slightly different terminology. We will adopt the terminology specified by the IETF IPPM RFC2330 [6], because it is a reasonably well-defined specification agreed upon by a large number of people. Its origins are in the networking community, whereas many GGF participants are from other communities, but it is the most appropriate terminology in existence.

3.1 Metric

A metric is a quantity related to the performance and reliability of the Internet. More specifically, a metric is a primary characteristic of the Internet, or of the traffic on it. A metric is the characteristic itself, not an observation of that characteristic. An example of a metric is link capacity.

3.2 Measurement

A measurement is an observation of a metric. Generally, there will be multiple ways to measure a given metric. Measurements may be either “raw” or “derived.” Raw measurements are something that can be measured directly, such as measuring latency using ping, or measuring capacity via an SNMP query to a router. Derived measurements are measured indirectly, and might be an aggregation or estimation based on a set of low-level measurements, such as using a statistical analysis of bursts of packets to estimate available bandwidth (e.g.: pchar and pathrate).

Other examples of measurements, their relationship to particular metrics, and issues involved with different types of measurements are given below.

3.3 Metrics versus Measurements

To determine if a particular concept is a metric or a measurement, the most important factor is determining whether the technique used to make the observation has any influence on the value itself. In particular, if there are different ways to observe identical or almost identical concepts, resulting in different values, then the concept may be a metric, but the techniques are measurement methodologies.

To extend the link capacity example further, consider the question of whether TCP-capacity is a metric or a measurement. Although the maximum bandwidth that a TCP connection can achieve over a particular link is very important to many applications, it is not truly a metric. In particular it is a function of the link capacity, the TCP implementations used by both machines, and the power of the machines at each endpoint. Thus, link capacity is again the true metric, while the other parameters determine how the application-level observation relates to the link capacity metric.

(Aside: We are stating these distinctions more strongly than the original IPPM document. This is partially due to our desire to develop precise relationships, perhaps a hierarchy, between the various metrics and measurements, rather than simply establishing a flat dictionary of terms. Furthermore, discussions with IPPM contributors have indicated some questions as to what the differences between metrics and measurement methodologies are. While one difference between the NMWG and the IPPM is our focus in cataloging measurement methodologies as much as metrics, we wish to preserve a line between them when we describe them.)

4 Statistical Representations

There are several issues in determining the statistical representations for network metrics. Most observations are made as either packet traces or periodic samples of a particular metric. In either case, most applications do not use the actual sequence of measurements, but instead rely on some sort of statistical representation of those observations. This section will detail different representations that are in use.

4.1 Sampling Techniques

Following standard terminology, a single observation of a measurement is referred to as a *singleton.* Because there is little interest in single measurements, typically measurements are collected over a period of time at particular intervals. Such a series of measurements is referred to as a *sample.* A statistical measurement is a representation of a metric derived from a sample of measurements.

The first issue in building a representation is the sampling pattern used to collect individual observations of the metrics. Even for packet traces, while they capture all details during the trace, for most networks it is impossible to gather those traces continuously. Therefore, for either technique the interval at which the observations are made must be specified.

A number of techniques are in common use:

- Periodic intervals, beginning each observation at a consistent interval
- Aperiodic intervals, typically distributed according to a Poisson or geometric distribution.
- many more...

There are positives and negatives associated with each sampling technique.

4.2 Representing variability

Once the sampling technique has been determined, the next issue is how to represent the varying values observed for the metric at each interval. Although the simplest technique is to provide the sample in raw time series form, most customers of this information desire higher-level information.

Statistical representations span a wide range. Eventually, this section should classify and enumerate them for use in describing the information provided by tools. Simple techniques, such as mean and variance, may capture important information, but are not necessarily appropriate for all data. More detailed representations might use quartiles or percentiles to represent the entire distribution of measurements. More advanced representations such as wavelets are more complex than classical statistics, but capture the temporal component associated with many network behaviors.

5 Bandwidth Metrics

Bandwidth defined most generally as data per unit time. However, the “bandwidth of a link” is not a precisely defined term and specific metrics must be defined prior to discussing how to measure bandwidth.

5.1 Metrics

There are three metrics that describe bandwidth:

Capacity: The maximum bandwidth a path can provide to an application when there is no competing traffic load (cross traffic)

Availability: The maximum throughput that the path can provide to an application, given the path’s current cross traffic load.

Utilization: The aggregate bandwidth currently used by all applications on that path.

Each of these metrics describes characteristics of an entire path in addition to hop-by-hop behavior.

5.1.1 Application-level Bandwidth

The bandwidth metrics listed above are at a very low level. In fact, they are independent of whether the traffic is TCP or UDP. The IETF IPPM defines a TCP measurement of the available bandwidth metric: ”Bulk Transfer Capacity” (BTC) [5]. The specific definition of the bulk transfer capacity is:

$$\mathrm{BTC} = \mathrm{datasent}/\mathrm{elapsedtime}$$

Due to differences in TCP implementations and the fact the TCP standard allows for different congestion control algorithms, this is a very difficult metric to accurately measure. We define this as a measurement methodology, not a specific metric, and believe the IPPM is likely to follow in the near future.

In this section, we define two TCP metrics, although there are many others, as well as UDP metrics that need to be addressed in this section.

5.2 Issues

There are several issues associated with bandwidth measurement tools:

Intrusiveness: Some of these tools, such as iperf, can be quite intrusive. Experiments at SLAC [ref] have shown that to get a reasonable estimation of available bandwidth on a WAN using iperf requires about a 10 second test, which places a lot of unnecessary traffic on the network.

Accuracy: Tools like pathload and pipechar, which use packet train techniques to statically estimate the available bandwidth instead of actually measuring it directly are much less intrusive, but are often much less accurate. Ideally the tools should try to determine a "confidence" value associated with each measurement. There is also some debate whether packet train techniques can provide useful measurements of available bandwidth, or if they should be used only for capacity measurements.

Timeliness: All these tools take some time to run, so if an application or service needs this information it must either wait for a new test to run, or use the results of the previous test. Depending on the path dynamics, previous results may not be a good indication of future bandwidth. Applying time-series models is one way of determining the predictability of network traffic.

TCP Implementation: As is described in the BTC RFC, the TCP implementation can have a large influence on the achieved bandwidth. Any metric that relies on a system's TCP implementation is therefore subject to its influence on its results. Furthermore, tuning at the local machine, such as selecting the appropriate sized socket buffers, can have a profound influence on performance.

5.3 TCP-based measurements

TCP-based measurements are used to determine the maximum bandwidth available to a TCP connection. Because the measurement itself is the process of transferring data via TCP, it is perhaps the most accurate estimate of the bandwidth available to TCP between a particular pair of machines.

In addition to the implementation issues discussed above, the way in which TCP is used can seriously affect the results of the measurement. Applications that are frequently in slow-start, either because they open new TCP connections or because they frequently let the connection go idle, will observe different BTC than applications that send a steady stream of data. Some tools report only steady-state bandwidth, but an application must be aware of what portion of the TCP performance curve dominates its performance.

Finally, the relationship of TCP-based measurements to non-TCP applications, such as many multimedia applications, is highly questionable.

5.3.1 General TCP-bandwidth

A general form of a TCP measurement is to report the bandwidth obtained over a particular interval. This general representation allows the representation of a brief, standalone connection, as well as the representation of a small portion of an in-progress connection (such as provided by iperf).

Parameters:	
source	IP address of data source
destination	IP address of data sink
state	Can be from TCP establishment or part of an ongoing connection
source implementation	OS-specific implementation, ex. Linux 2.4.12
dest implementation	OS-specific implementation, ex. Linux 2.4.12
Data:	
length	Length of data transferred, in bytes
time	Beginning and ending times of measurement

5.3.2 TCP file transfer

This metric measures the time it takes for TCP to transfer a particular amount of data. Although it is strictly a subclass of the previous method, samples consisting of repeated transfers of the same length of data have somewhat different characteristics than observing the bandwidth of an ongoing connection at one-second intervals. For example, file transfers are typically, but not always, measured with a new TCP connection. Depending on the length of the file, it may or may not reach maximum bandwidth. Fixing the length of data transferred, rather than the time allowed for the transfer, can have significant statistical implications. Because these measurements almost always includes both a slow-start and steady-state phase, it can be difficult to extrapolate the time required to transfer files of different lengths.

Parameters:	
source	IP address of data source
destination	IP address of data sink
state	Can be from TCP establishment or part of an ongoing connection
source implementation	OS-specific implementation, ex. Linux 2.4.12
dest implementation	OS-specific implementation, ex. Linux 2.4.12
length	Length of data being transferred.
Data:	
time	Beginning and ending times of file transfer

5.4 Packet dispersion measurements

Packet train, packet pair, or packet dispersion techniques work by generating traffic consisting of back-to-back packets sent from source to destination. By measuring the difference between the separation of the packets at the source and destination, referred to as the dispersion, characteristics of the path are inferred.

There is general consensus that packet dispersion metrics are well-suited for measuring path capacity. Packet dispersion metrics are also used for measuring available bandwidth. There is, however, some question about their ability to measure availability in all situations [2], and even in the ideal situations, achieving an accurate result may require many measurements.

Parameters:	
source	IP address of data source
destination	IP address of data sink
source implementation	OS and software used for the measurement
dest implementation	OS and software used for the measurement
packets	Number of consecutive packets in a single burst
length	Length of each packet in the burst
Data:	
time	Arrival time of each packet

6 Delay Metrics

general definition of delay and issues. discussion of one-way vs roundtrip. reference discussions on clock synchronization

6.1 One-way delay measurements

6.2 Roundtrip delay measurements

6.3 Delay jitter measurements

6.4 Issues

7 Jitter Metric

For example, iperf in UDP mode measures this.

7.1 Issues

8 Loss Metric

ping can be used for this, but has classification issues. UDP probes. TCP (from kernel level).

8.1 Issues

9 Tools

The following is a list of a few tools and the metrics, as defined above, that each tool provides. This is not meant to be a complete list. The goal is to create a web page with a more complete and current list of tools.

9.1 iperf

- Metrics: available bandwidth, jitter (in UDP mode)
- Pros:
- Cons:
- Related tools: ttcp, nettest

9.2 ping

- Metrics: delay, loss
- Pros:
- Cons:
- Related tools:

9.3 pipechar

- Metrics: hop-by-hop capacity, available bandwidth, loss
- Pros:
- Cons:
- Related tools: pchar, pathrate

9.4 NWS

- Metrics:
- Pros:
- Cons:
- Related tools:

10 Full Copyright Notice

11 Intellectual Property Rights

The GGF takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the GGF Secretariat.

The GGF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to practice this recommendation. Please address the information to the GGF Executive Director.

12 References

References

[1] Ann Chervenak and GGF: Data Replication Research Group. An architecture for replica management in grid computing environments. http://www.sdsc.edu/GridForum/RemoteData/Papers/ggf1replica.pdf.

[2] Constantinos Dovrolis, Parameswaran Ramanathan, and David Moore. What do packet dispersion techniques measure? In *IEEE INFOCOM 2001*, 2001.

[3] GGF. Grid scheduling area. http://www.mcs.anl.gov/˜schopf/ggf-sched.

[4] T. J. Hacker and B. D. Athey. The end-to-end performance effects of parallel tcp sockets on a lossy wide-area network.

[5] M. Mathis and M. Allman. A framework for defining empirical bulk transfer capacity metrics. RFC3148, July 2001.

[6] V. Paxson, G. Almes, J. Mahdavi, and M. Mathis. Framework for IP performance metrics. RFC2330, May 1998.